<u>REMARKS</u>

This application has been reviewed in light of the Office Action dated July 25, 2008. Claims 1-14 are presented for examination, of which Claims 1, 7 and 12 are in independent form and have been amended to define still more clearly what Applicant regards as his invention. Favorable reconsideration is respectfully requested.

In the outstanding Office Action, Claims 1-14 were rejected under 35 U.S.C. § 103(a) as being obvious from U.S. Patents 6,853,465 (Ohnishi) and 6,490,055 (Shimizu), taken in combination.

As discussed in the specification and previous Amendments, there are several conventional types of color image processing by printer drivers, which receive rendering instructions and produce binary bitmaps for printing engines. One is the high-quality mode consisting of 1) multivalue rendering, 2) color correction, 3) color conversion, and 4) binarization in that order (Fig. 3) except that 2) may also be performed before 1) (paragraph [0029]). Another is the CMYK high-speed mode consisting of 1) color correction, 2) color conversion, 3) binarization, and 4) binary rendering in that order (Fig. 5).

In the high-quality mode, a multivalued bitmap is developed from Step 1, and the entire bitmap is subject to color processing in Step 3 and possibly Step 2. As a result, a large memory is required (paragraph [0005]). In addition, color processing and binarization in Step 4 are done pixel by pixel, resulting in low processing efficiency.

On the other hand, when an input rendering instruction involves anything other than a simple overwrite, the application of the CMYK high-speed mode is very likely

to result in poor output quality, as no linear-independence requirement of the rendering instruction can be met in the CMYK color space (paragraph [0007]).

Therefore, each mode has its merits and drawbacks and should be applied as appropriate. The main difference between the two modes is that in the high-quality mode, color conversion (Step 3) and binarization (Step 4) are performed after bitmap development (Step 1), while in the CMYK high-speed mode, color conversion (Step 2) and binarization (Step 3) are performed before bitmap development (Step 4).

Furthermore, in a scanline rendering paradigm, rendering is performed on a row-by-row basis to achieve high efficiency (as opposed to a pixel-by-pixel basis) and low memory usage (as opposed to a polygon-by-polygon basis). Note that each rendering instruction may cover multiple scanlines, and each scanline may be covered by multiple rendering instructions (paragraph [0055]). In examining each scanline, it would be important to identify all the rendering instructions that cover the scanline.

The present invention relates to a printing control method that further increases the efficiency of typical scanline rendering by processing scanlines in different modes depending on the nature of rendering instructions. For each scanline, if all the rendering instructions covering this scanline are for overwrites, a bitmap will be developed for the scanline in a generalized CMYK high-speed mode, where Step 3 and Step 4 are performed with n values, with n=2 being binarization and binary rendering, respectively (S7 in Fig. 8). As long as one of the rendering instructions involves anything other than an overwrite, however, a bitmap will be developed for the scanline in a generalized high-quality mode, where Step 4 is similarly performed with n values.

Claim 1 recites, among other features, "first rendering means for developing rendering instructions of each scan line into multivalued bitmap data [multivalue rendering] and subjecting the multivalued bitmap data to color processing [color correction and color conversion] and n-value conversion processing [binarization, for example]; second rendering means for subjecting the rendering instructions to color processing [color correction and color conversion] and n-value conversion processing color by color of the rendering instructions, storing the results in the form of an n-valued pattern [binarization, for example], and rendering the n-valued pattern of each line into n-valued bitmap data [binary rendering, for example]; ... and control means for... exercising control so as to cause said first rendering means to form the multivalued bitmap data... if said determining means determines that the rendering instructions include a rendering instruction other than the overwrite, and to cause said second rendering means to form the n-valued bitmap data if said determining means determines that the rendering instructions do not include a rendering instruction other than the overwrite, wherein said control means causes said first rendering means or said second rendering means to develop all the rendering instructions for one scan line into bitmap data before rendering the next scan line."

Applicant notes that the first rendering means of Claim 1 applies the generalized high-quality mode one scan line at a time, and the seconding rendering means applies the generalized CMYK high-speed mode where the last step – binary rendering, for example – is done one scan line at a time.

The second rendering means is not believed to be disclosed or taught in *Ohnishi*. The portion of *Ohnishi* cited in the Office Action as disclosing the second rendering means is as follows:

> While referring to the pattern plane, *color conversion*, which is consonant with the attribute of an object, is performed for the obtained multi-value bit map (*multivalue rendering*), and the resultant bit map is binarized (n-valued) (*binarization*) to obtain a device bit map. When the processing has been completed for the overall image, the device bit map is transmitted to the printer. *Color correction* may be performed either before or after the color data have been used to generate the bit map [parenthesized, italicizations and emphasis added]. Col. 4, lines 15-22.

Thus, in *Ohnishi*, multivalue rendering is performed, color conversion is performed, and binarization is performed; color correction may be performed before or after multivalue rendering. This is exactly how the high-quality mode works, as discussed above in these Remarks. In particular, color conversion and binarization are performed after (rather than before) bitmap development.

Since it does not disclose the second rendering means of Claim 1, *Ohnishi* also does not disclose the control means, which depends on the second rendering means.

Shimizu does not remedy these deficiencies, either. *Shimizu* relates to a method that performs multivalue rendering by processing one rendering instruction after another. The method uses hardware rending as well as software rending depending on the nature of rendering instructions. However, it does not perform rendering on a scanline basis by processing all the rendering instructions covering that scanline and developing a bitmap only for that scanline each time. Therefore, both the seconding rendering means and the control means of Claim 1 are missing from *Shimizu*.

Specifically, after it interprets all the rendering instructions for one page (*see* Fig. 2, of *Shimizu*), the method determines whether the page can be subjected to banding (*see* col. 7, lines 41-46). If the page can be subjected to banding, such as when each of the rendering instructions involves nothing more than a simple overwrite (*see* col.

8, lines 50-65), the method performs hardware rendering at full gradation (*see* col. 2, lines 10-25 and lines 33-69) – *multivalue rendering* – to achieve high output quality, one or more bands at a time (*see* Fig. 8) – certainly more than one scanline at a time (*see* col. 6, lines 38-42).

If the page cannot be subjected to banding, such as when any rendering instruction involves a logical operation (*see* col. 8, lines 50-65 and col. 10, lines 36-42), the method subjects the page to color logic drawing (*see* Fig. 10) and performs software *rendering* at lowered gradation (*see* col. 2, lines 43-52; col. 3, lines 10-15; col. 7, lines 3-7) – even if it is somehow referred to as high-grade logical drawing – *before color conversion* to the CMYK color space (*see* 406 and 408 in Fig. 9; col. 10, lines 43-51) – to achieve good output quality while lowering memory usage – for each rendering instruction rather than each scanline (*see* input object 501 in Fig. 10, which is converted from input PDL data/PDL command 101 and in Fig. 2; col. 5, lines 49-52 and col. 10, lines 1-3).

Therefore, *Shimizu* does not disclose "second rendering means for subjecting the rendering instructions to... n-value conversion processing (color conversion)... storing the results in the form of an n-valued pattern (binarization, for example), and rendering the n-valued pattern of each scan line into n-valued bitmap data (binary rendering, for example); ... and control means for... exercising control so as to... cause said second rendering means to form the n-valued bitmap data if said determining means determines that the rendering instructions do not include a rendering instruction other than the overwrite, wherein said control means causes said first rendering means or said second rendering means to develop all the rendering instructions for one scan line into bitmap data before rendering the next scan line [parenthesized expressions added]."

The portions of *Shimizu* cited in the Office Action as disclosing the control means are as follows:

> ... and control means for controlling the band rendering to be executed without lowering the color gradation when the banding process is judged to be executable by said judgement means, or the degrade rendering to be executed by lowering the color gradation when the banding process is judged to be inexecutable. Col. 3, lines 10-15, of *Shimizu.*

The paragraph above describes merely the control means in *Shimizu* that controls whether to perform banding (hardware rendering) at full gradation or color logic drawing (software rending) at lowered gradation.

> At step 901, the CPU 12 extracts scan line information (x min, x max) in the Y coordinate from the mask information of intermediate data created in the management RAM 7 through the PDL analysis task 120, and writes corresponding background information into a band raster memory 10 by referring to the current background information and logical drawing mode. Col. 8, lines 10-16, of *Shimizu.*

The paragraph above describes merely the preparation for rendering one or more bands. Various information for the band(s) is placed in the band raster memory. Using this information, rendering is performed for the band(s) and the result – a multivalued bitmap – is placed in the band memory (*see* 805 in Fig. 12; col. 11, lines 59-61) and eventually sent to the printer engine (col. 8, lines 28-33).

> In this embodiment, the mask information 151 to be supported is composed of run length (one scan line in the X direction), convex polygon with no edge crossed, bit map image, and bit map font. As can be seen from such information, the mask information is made a suitable structure for the fast hardware rendering, for example, a pentagon of FIGS. 5A to 5D is subdivided into five triangles not crossed, as shown in FIG. 5B, at step 104 (the painting is in accordance with an even-odd rule in this embodiment). In a line connection processing portion as shown in FIG. 5C, by applying a DDA algorithm in this module, information is expanded into a work area within the management RAM 7 in view of the line connection information (round, miter, triangle), and the final external shape is held in the run length manner for each Y scan line, with min x and max x as pair information, to

> prepare for the fast rendering to be subsequently performed. Col. 6, lines 21-37, of *Shimizu.*

The paragraph above describes merely how to create mask information in interpreting input rendering instructions or drawing commands. It again does not concern the actual rendering or bitmap development.

Accordingly, for at least the reasons noted above, Claim 1 is believed to be allowable over *Ohnishi* and *Shimizu*, taken separately or in any permissible combination (if any).

Independent Claims 7 and 12 are directed, respectively, to a method and a printer driver, and correspond to apparatus Claim 1, and are believed to be patentable for at least the same reasons as discussed above in connection with Claim 1.

A review of the other art of record has failed to reveal anything which, in Applicant's opinion, would remedy the deficiencies of the art discussed above, as references against the independent claims herein. Those claims are therefore believed patentable over the art of record.

The other claims in this application are each dependent from one or another of the independent claims discussed above and are therefore believed patentable for the same reasons. Since each dependent claim is also deemed to define an additional aspect of the invention, however, the individual reconsideration of the patentability of each on its own merits is respectfully requested.

In view of the foregoing amendments and remarks, Applicant respectfully requests favorable reconsideration and early passage to issue of the present application.

Applicant's undersigned attorney may be reached in our New York office by telephone at (212) 218-2100. All correspondence should continue to be directed to our below listed address.

Respectfully submitted,

/Leonard P Diana/
Leonard P. Diana
Attorney for Applicant
Registration No. 29,296

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3801
Facsimile: (212) 218-2200

FCHS_WS 2603659v1